

Windows Programming

Visual C++ MFC Programming

Lecture 11

김예진

Dept. of Game Software

Notices

- 03/07: 502 → 501 등록 이동
- 03/21: HW 1 (Due: 03/28)
- 04/09: HW 2 (Due: 04/16)
- 04/25: Midterm (실습, ~75 min, 강의록 1~7)
- 05/14: HW 3 (Due: 05/21)
- 06/04: HW 4 (Due: 06/11)
- 06/13: Final (실습, ~75 min., 강의록 전체)
- 06/18: 보강
 - 501: 13:00~15:00
 - 502: 15:00~17:00

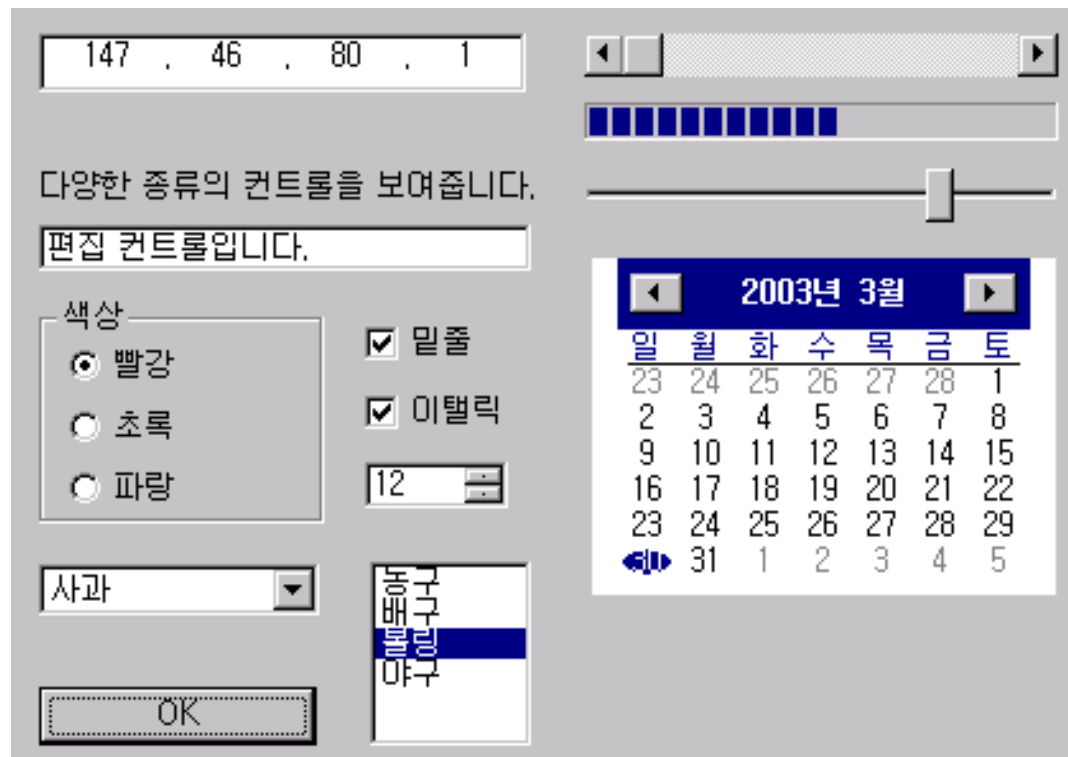
Plan

- Controls
 - Buttons
 - Check Box
 - Dialog Box
 - Static/Edit
 - List/Combo Box
 - Scroll Bar



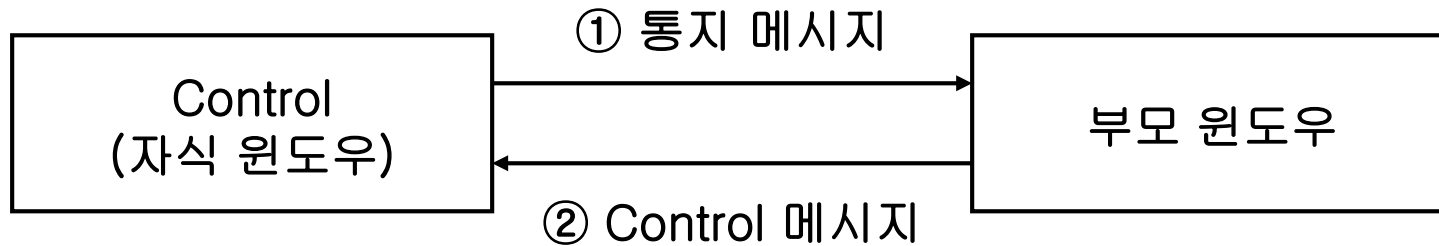
Controls

- Control Windows
 - 표준화된 형태와 특성을 가진 윈도우
 - 사용자에게 입력을 받거나 정보를 보여줌



Controls

- Control messages

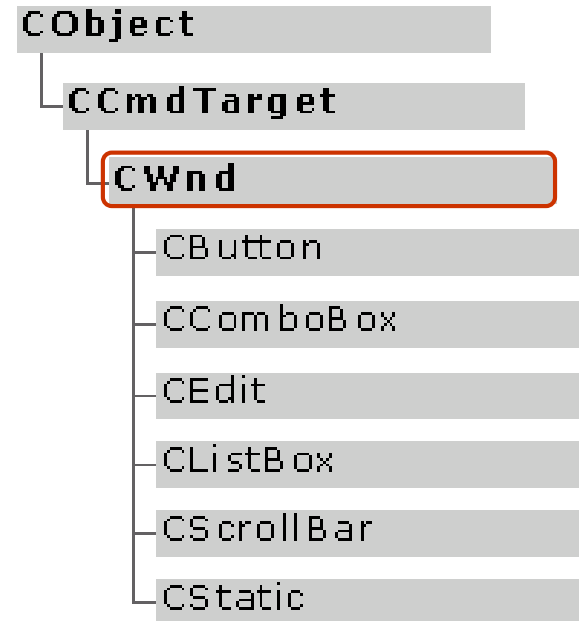


- 통지(notification) message
 - Control의 상태가 변화되었음을 알림
 - Memory 부족 등으로 인한 오류를 알림
- Control message
 - Control의 상태를 알아내거나 변경

Controls

- MFC control class
 - Control도 윈도우의 일종임

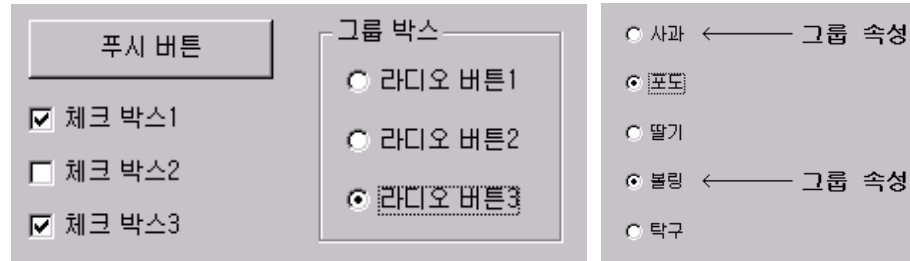
Control	MFC Class	Control Button
Button	CButton	
Static	CStatic	
Edit	CEdit	
List Box	CListBox	
Combo Box	CComboBox	
Scroll Bar	CScrollBar	



- Control 생성 방법
 - 방법 1: Code 직접 추가
 - 방법 2: 대화 상자 resource를 통해 생성

Controls

- Button 생성



- Code로 직접 추가

- 1. Button object를 member 변수로 추가

```
// ChildView.h
CButton m_button;
```

- 2. CButton::Create()을 호출하여 생성

```
// CChildView::OnCreate() 내부에 추가
m_button.Create(_T("누르세요"),           // Caption 문자열
    WS_CHILD | WS_VISIBLE | BS_PUSHBUTTON, // 윈도우/버튼 style
    CRect(100, 100, 200, 130),           // 크기와 위치
    this,                                 // 부모 윈도우
    101);                                 // ID
```

Button Style	의미
BS_PUSHBUTTON	Push button
BS_CHECKBOX	Check box
BS_3STATE	3상태 check box
BS_RADIOBUTTON	Radio button
BS_GROUPBOX	Group box

Controls

- Button의 동작

- Button을 누르면 눌렀다는 통지 message 발생



- 통지 message 처리

1. Message map에 message 추가

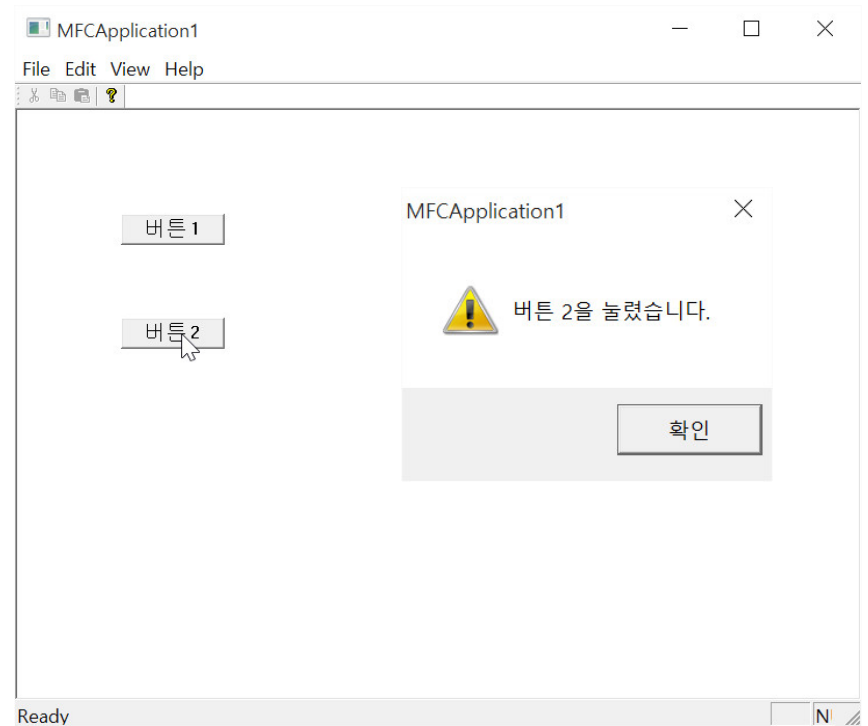
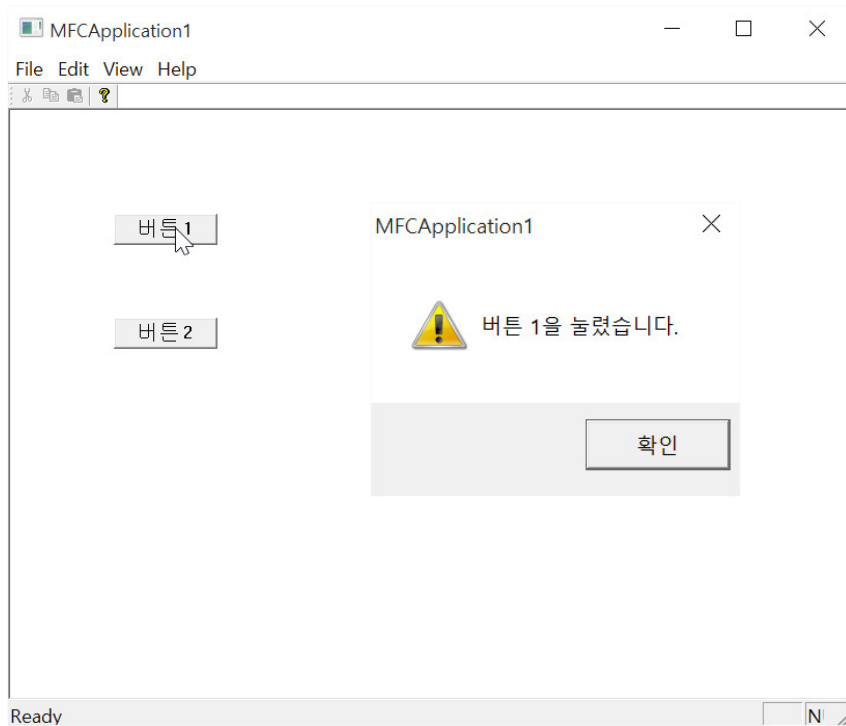
```
// CChildView.cpp
BEGIN_MESSAGE_MAP(CChildView, CWnd)
...
    ON_BN_CLICKED(101, OnButtonClicked)    // Control ID, message handler 함수
END_MESSAGE_MAP()
```

2. Message handler 함수 구현

```
void CChildView::OnButtonClicked()
{
    AfxMessageBox(_T("버튼을 눌렀습니다."));
}
```


Controls

- 연습: Push button 붙이기
 - Push button 2개를 사용자 영역에 생성
 - 어떤 button이 눌러졌는지 화면에 표시



Controls

- Button의 정보 변경

- Button의 상태 변경시 Control 메시지 활용



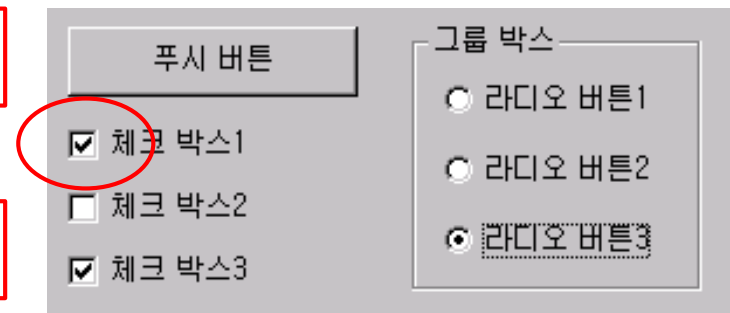
- Control 메시지 보내기

- Button의 현재 상태 변환

```
m_button.SetCheck(BST_CHECKED);  
m_button.SetCheck(BST_UNCHECKED);
```

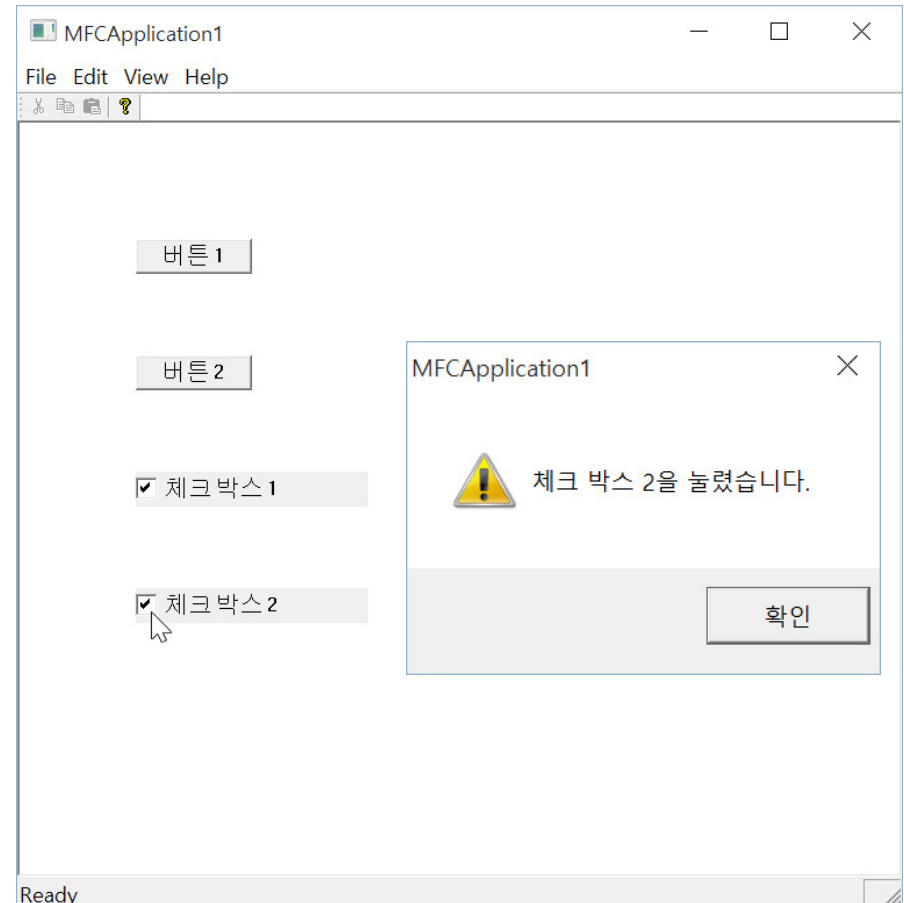
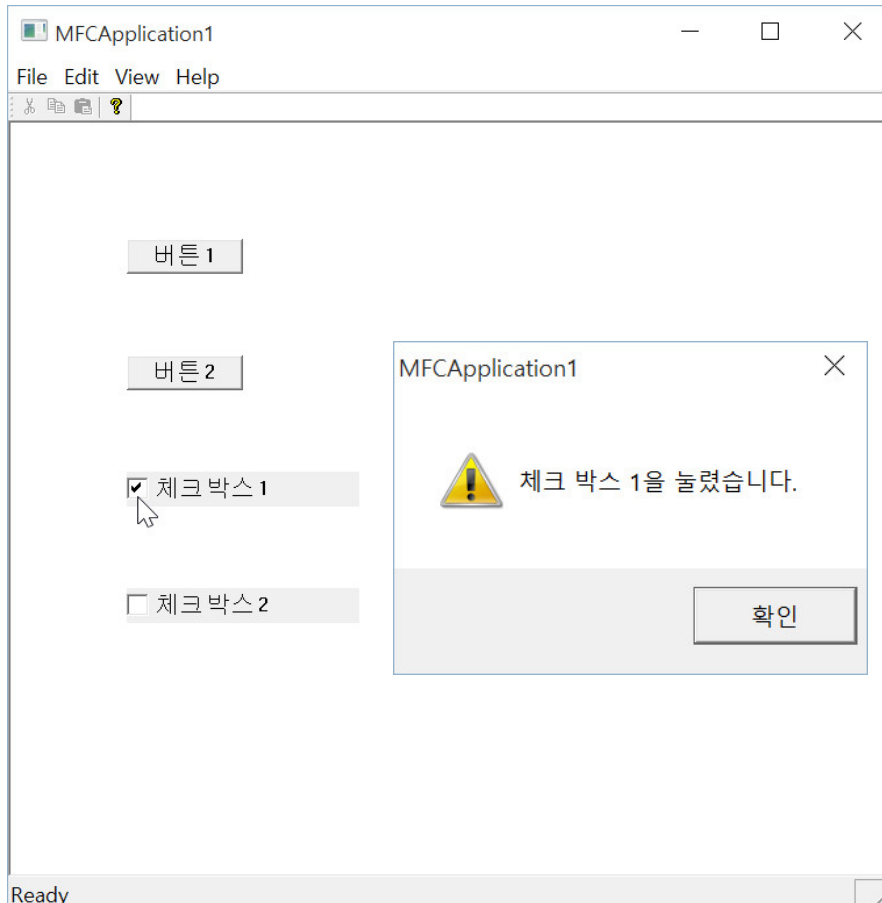
- Button의 현재 상태 확인

```
if (m_button.GetCheck() == BST_CHECKED)  
    AfxMessageBox(_T("버튼이 체크되었습니다.));
```



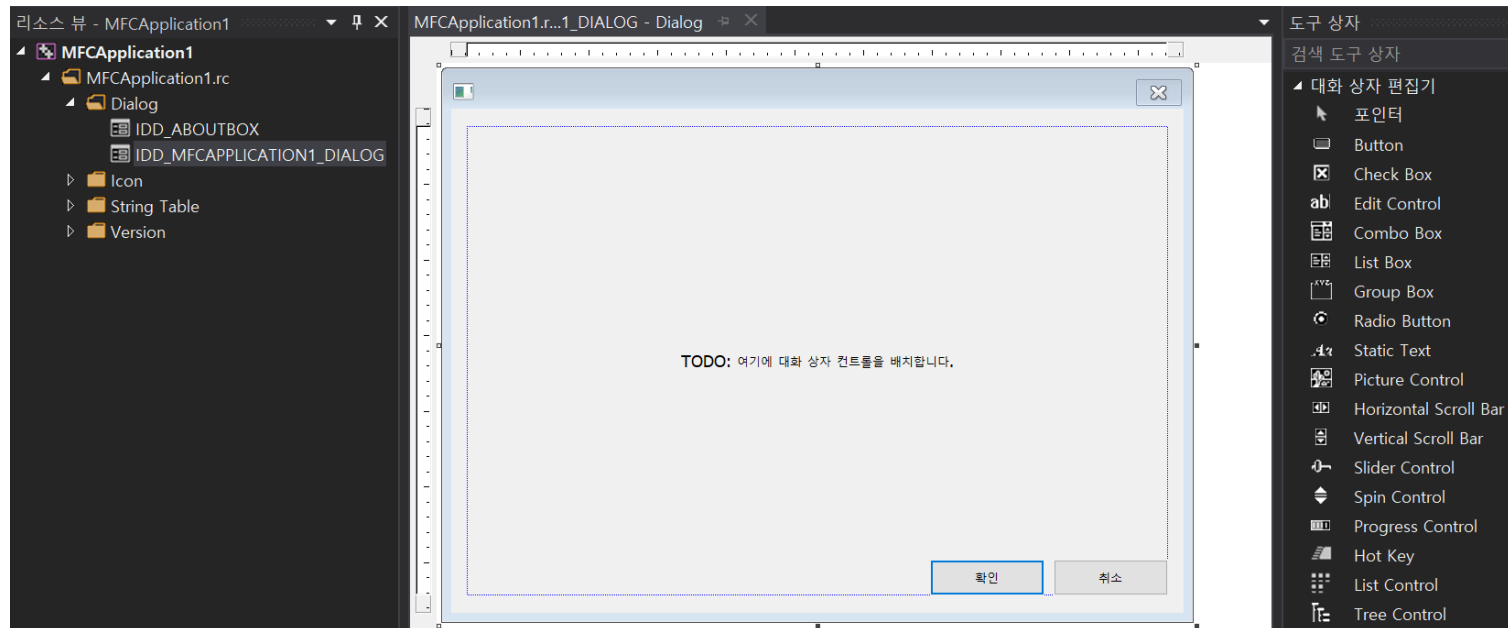
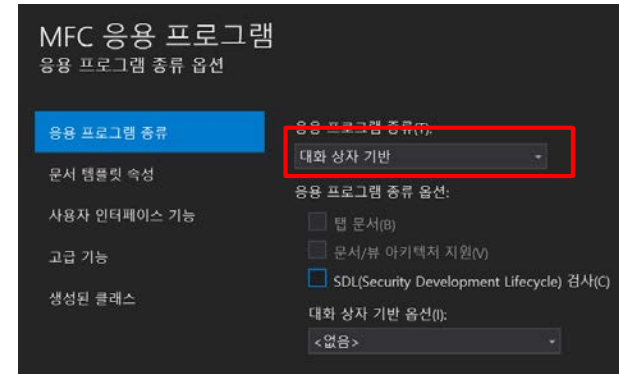
Controls

- 연습: Check box 붙이기
 - Check box 2개를 사용자 영역에 생성
 - 어떤 box가 눌러졌는지 화면에 표시



Controls

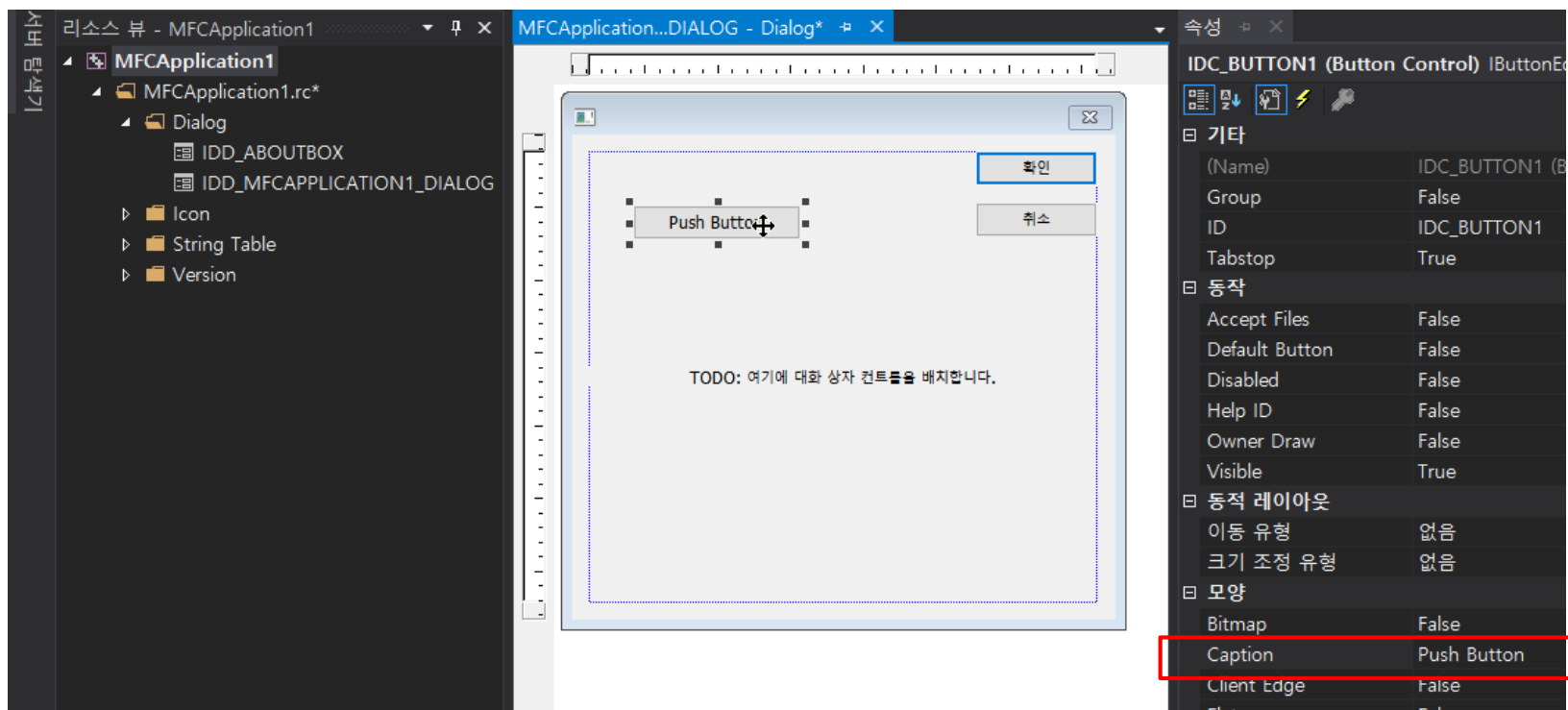
- Control 생성: Dialog Box(대화 상자) 사용
 - MFC 응용 프로그램 → 응용 프로그램 종류 → 대화 상자 기반
 - 대화 상자에 Control 추가
 - Control 변수 생성
 - Control event handler 생성



Controls

- Control 속성 변경

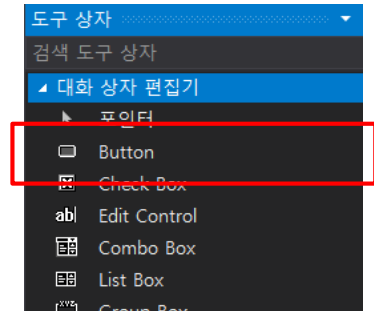
1. 대화 상자에 Control 추가 (예: Button)
2. 추가한 Control 위치 및 크기 수정
3. 추가한 Control 선택 → [RMB] 속성 → 변경 (예: Caption)



Controls

- Control 변수 생성

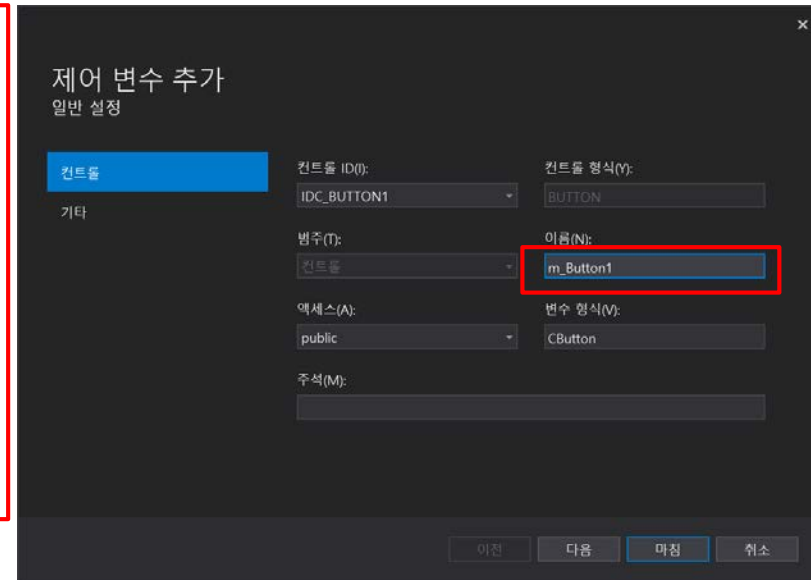
1. 도구 상자 → Control 추가 (예: Button)
2. 추가한 Control 선택 → [RMB] 변수 추가
3. 제어 변수 추가 → 컨트롤 → 이름 (예: m_Button1)



```
// MFCApplication1Dlg.h
class CMFCApplication1Dlg : public CDlglogEx
{
    ...

public:
    CButton m_Button1;
}

// MFCApplication1Dlg.cpp
void CMFCApplication1Dlg::DoDataExchange(CDataExchange* pDX)
{
    CDialogEx::DoDataExchange(pDX);
    DDX_Control(pDX, IDC_BUTTON1, m_Button1);
}
```

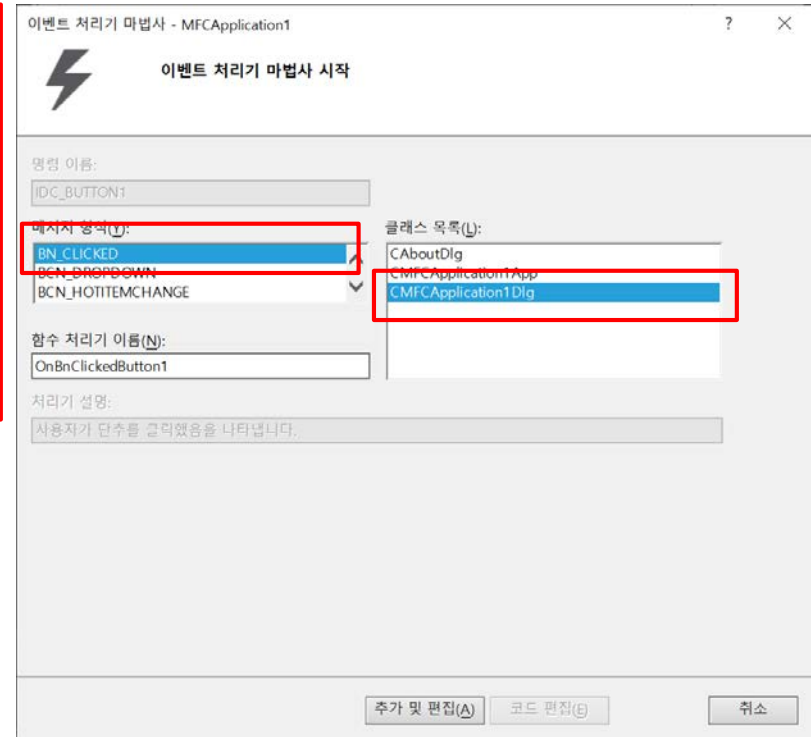


Controls

- Control event handler 추가
 - 추가한 Control 선택 → [RMB] 이벤트 처리기 추가
 - 이벤트 처리기 마법사 → 메시지 형식 선택 (예: BN_CLICKED)
 - Event handler 함수 (예: OnBnClickedButton1()) 수정

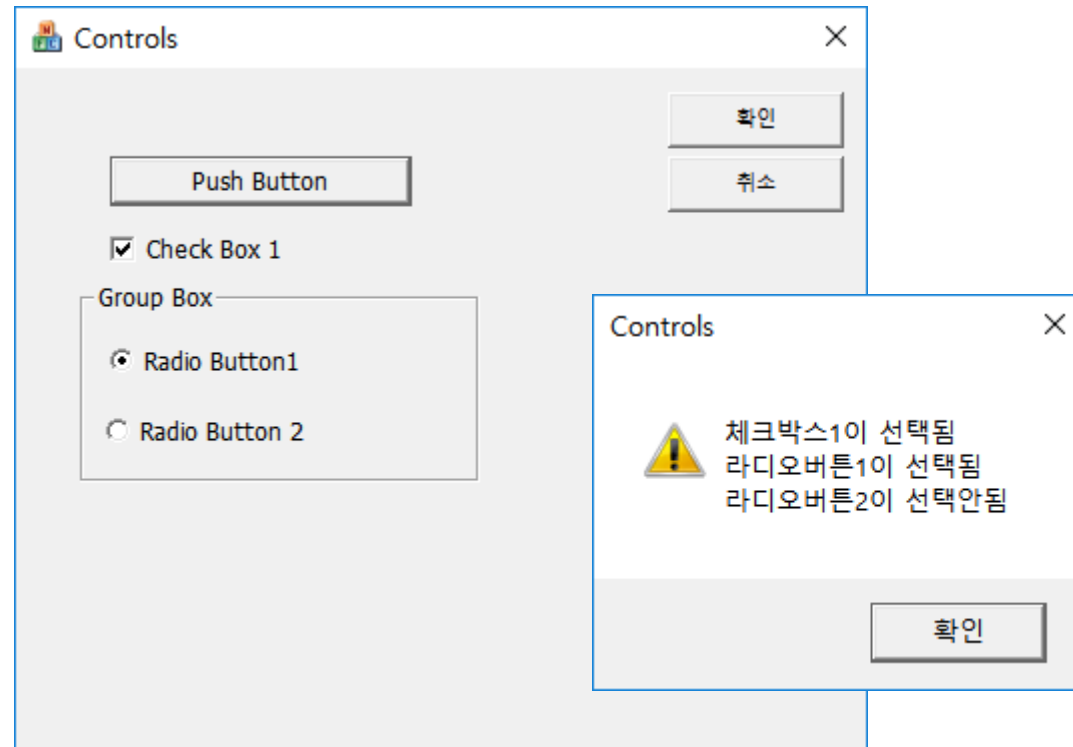
```
// MFCAplication1Dlg.cpp
void CMFCAplication1Dlg::OnBnClickedButton1()
{
    // TODO: 여기에 컨트롤 알림 처리기 코드를 추가합니다.
    CString str;
    if (m_Button1.GetCheck())
        str = _T("체크 박스1이 선택됨");
    else
        str = _T("체크 박스1이 선택 안됨");

    AfxMessageBox(str);
}
```



Controls

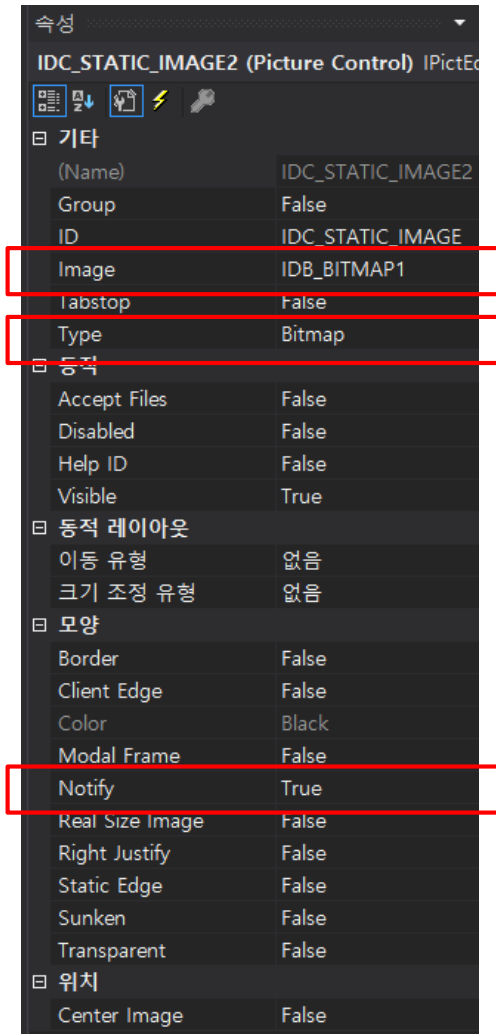
- 연습: 대화 상자를 통한 다양한 버튼 생성
 - Push button, check(1개) & radio(2개) button을 대화 상자에 추가
 - Push button을 누르면 현재 button들의 상태를 표시



Controls

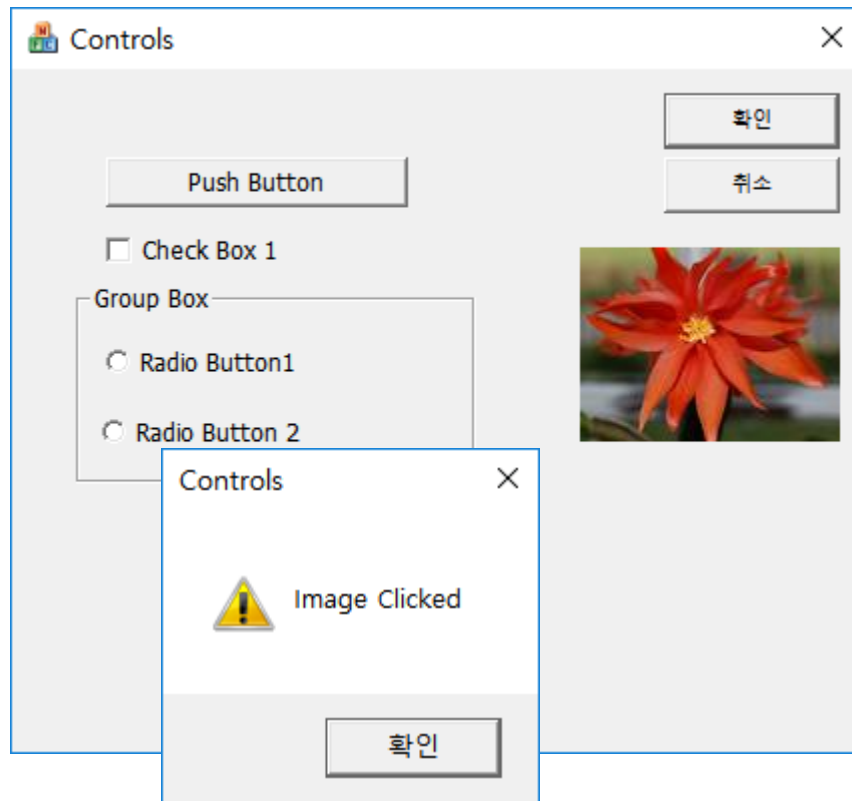
- Static: Text/Picture 출력
 - Static Text: 일반 문자열 출력
 - Picture Control: 그림 출력
 - Type: Bitmap
 - Image: Bitmap 생성 후 ID 지정
 - Button 기능 추가
 - Notify: True → 통지 message 발생
 - 클릭: STN_CLICKED
 - 더블 클릭: STN_DBLCLK
 - 해당 통지 message handler 추가

```
void CMFCApplication1Dlg::OnStnClickedStaticImage()  
{  
    AfxMessageBox(_T("Image Clicked"));  
}
```



Controls

- 연습: Picture control 추가
 - Picture control을 대화 상자에 추가
 - Bitmap을 picture control 연결하여 대화 상자에 그림 출력
 - 그림 click시 눌렀다는 메세지 출력

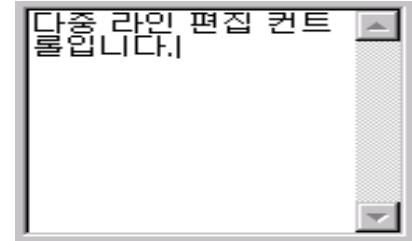


Controls

- Edit control

- 단일 또는 다중 라인 편집
- 주요 통지 messages

단일 라인 편집 컨트롤



Message

의미

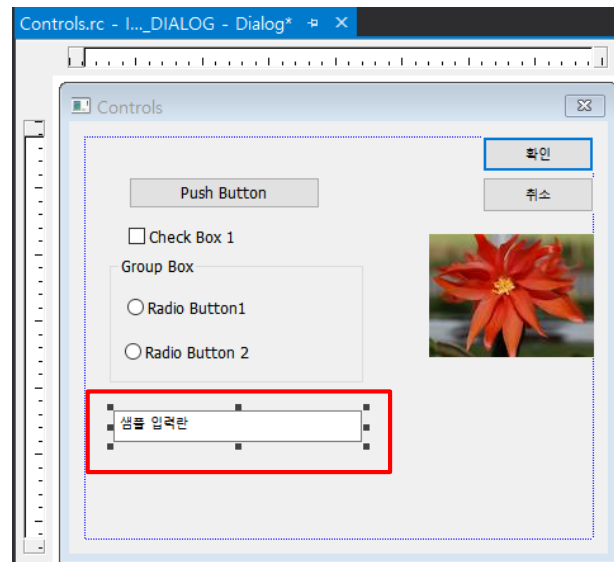
EN_CHANGE

사용자가 control의 내용을 변경하면 화면에 control을 다시 그린 후 발생

EN_UPDATE

사용자가 control의 내용을 변경하면 화면에 control을 다시 그리기 직전에 발생

- Edit control 추가



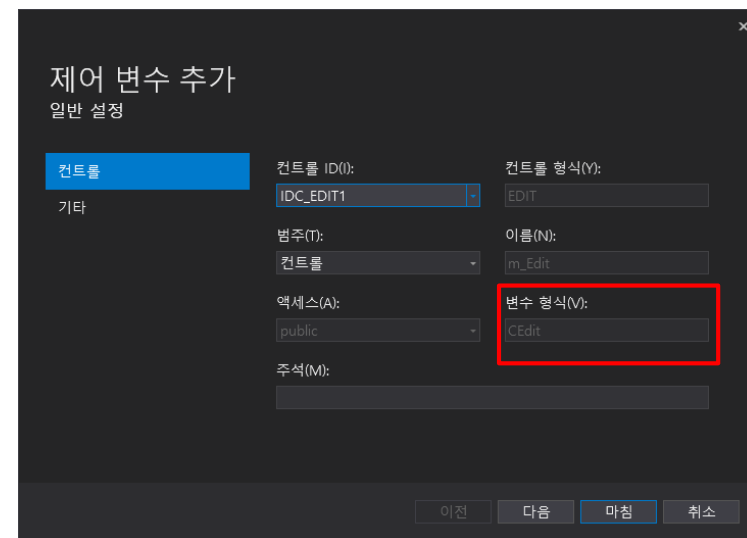
Controls

- Edit control 제어

- Edit text의 상태를 편집시 Control 메시지 활용



- Control 메시지를 전달하기 위해서는 CEdit type 변수 추가
 - 방법 1: DoDataExchange() 함수 이용한 직접 연결
 - 방법 2: Control의 변수 추가 기능 이용
- Control message handler 추가



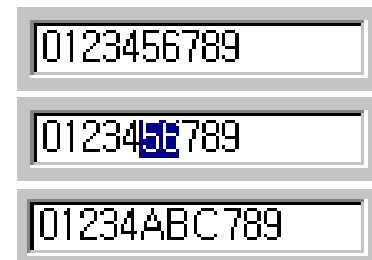
Controls

- Edit control 제어
 - Text를 변경하거나 입력된 text를 알아내기

```
m_Edit.SetWindowText(_T("초기값입니다.));  
CString str;  
m_Edit.GetWindowText(str);
```

- 텍스트 선택과 치환하기

```
m_Edit.SetSel(5, 7);  
m_Edit.ReplaceSel(_T("ABC"));
```



- 입력 가능한 문자열의 길이를 제한

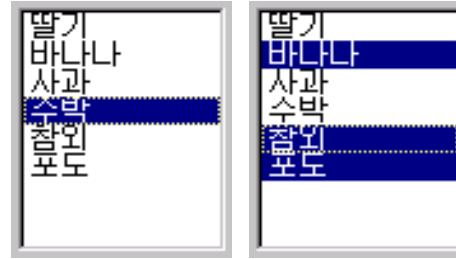
```
m_Edit.SetLimitText(10);
```

- 각종 편집 작업

```
m_edit.Clear();  
m_edit.Cut();  
m_edit.Copy();  
m_edit.Paste();  
m_edit.Undo();
```

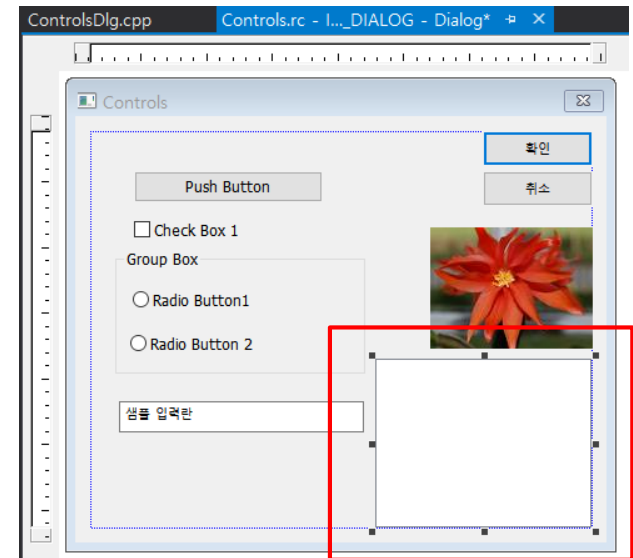
Controls

- List box
 - 단일 또는 다중 라인 선택
 - 주요 통지 messages



Message	의미
LBN_DBLCLK	사용자가 항목을 더블 클릭
LBN_SELCHANGE	사용자가 선택을 변경
LBN_SELCANCEL	사용자가 선택을 취소

- List box 추가 및 제어
 - 통지 message 발생: 속성 → Notify: True
 - CListBox type 변수 추가
 - Control message handler 추가



Controls

- List box 제어
 - List 항목 추가와 삭제

```
m_List.AddString("사과");  
m_List.DeleteString(3);
```

- List 항목 선택

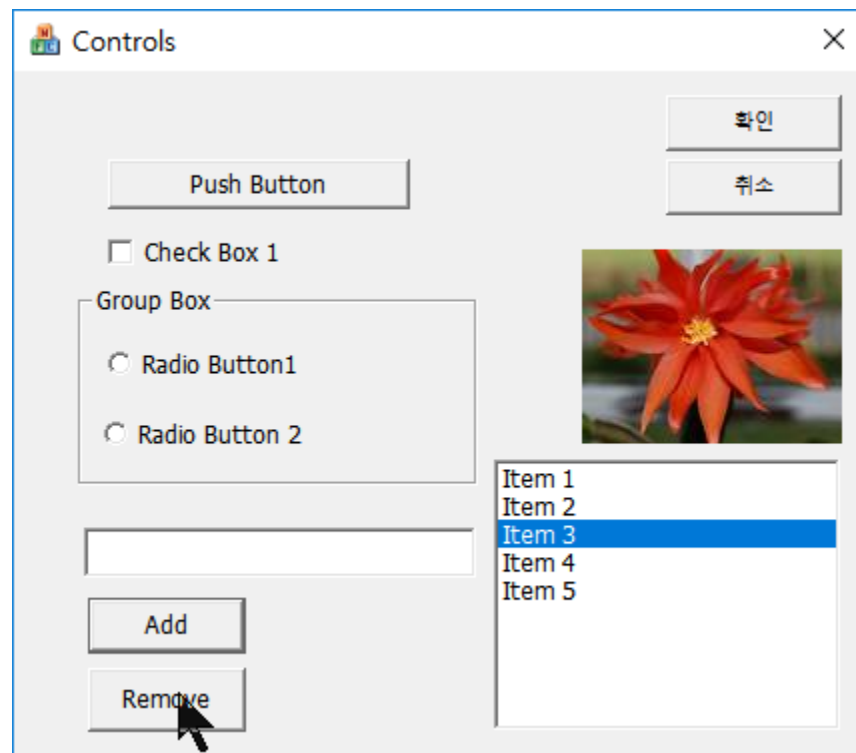
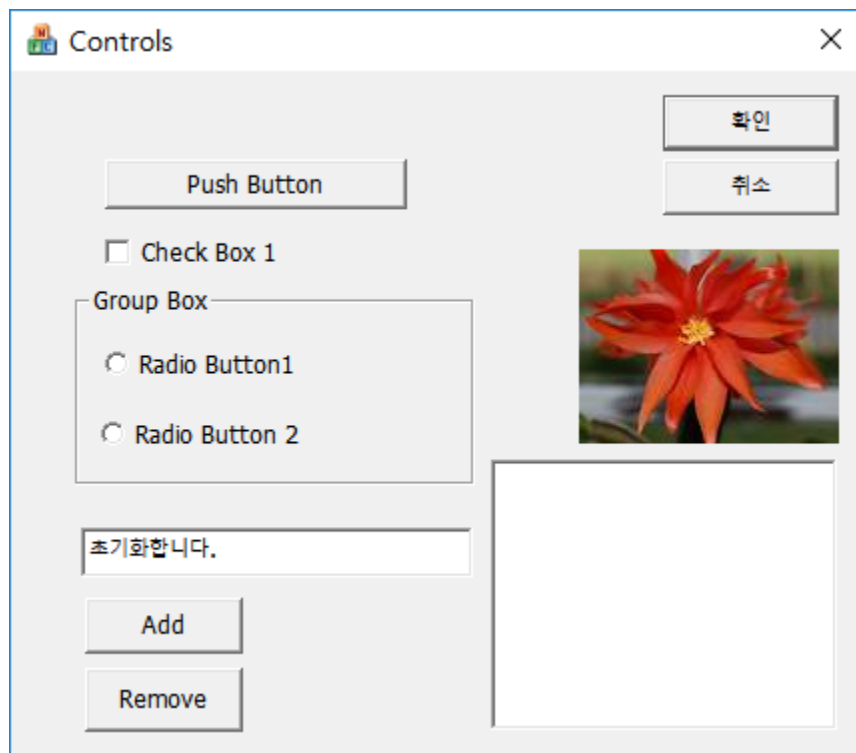
```
m_List.SetCurSel(2); // 단일 선택 리스트 박스 컨트롤인 경우  
// 다중 선택 리스트 박스 컨트롤인 경우  
m_List.SetSel(2);  
m_List.SetSel(3, FALSE);
```

- 선택한 list 항목 알아내기

```
// 단일 선택 리스트 박스 컨트롤인 경우  
int nIndex = m_List.GetCurSel();  
If (nIndex != LB_ERR) {  
    CString str;  
    m_List.GetText(nIndex, str);  
}  
  
// 다중 선택 리스트 박스 컨트롤인 경우  
int nIndex = m_List.GetCaretIndex();  
If (nIndex != LB_ERR) {  
    CString str;  
    m_List.GetText(nIndex, str);  
}
```

Controls

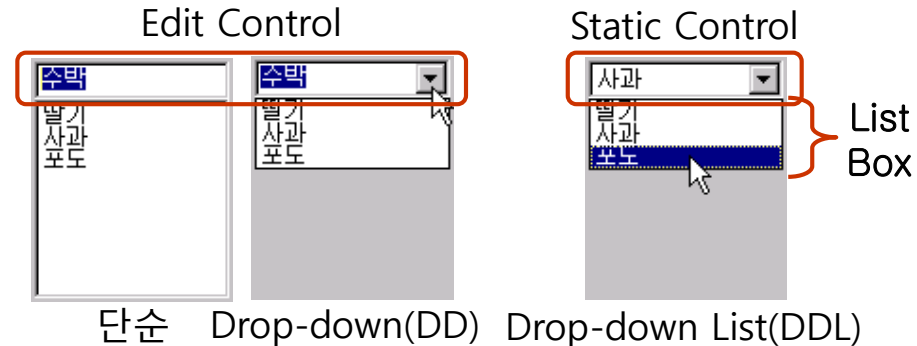
- 연습: List Box 항목 추가 및 삭제
 - List Box를 대화 상자에 추가
 - Edit를 통해 입력한 text를 List Box 항목에 추가 또는 삭제
 - List Box 항목 추가 및 삭제를 위해 Push Button 2개 추가



Controls

- Combo Box

- 여러 Control을 같이 사용
- Style: 단순, DD, DDL
- 주요 통지 messages



Message	의미	단순	DD	DDL
CBN_SELENDOK	사용자가 항목을 선택	•	•	•
CBN_DROPDOWN	List Box가 열리기 직전		•	•
CBN_CLOSEUP	List Box가 닫혔다.		•	•
CBN_EDITCHANGE	사용자가 Edit의 문자를 변경	•	•	
CBN_EDITUPDATE	변경된 문자를 Edit에 출력하기 직전	•	•	

- 항목 추가와 삭제

```
m_combo.AddString(_T("사과"));
m_combo.DeleteString(3);
```

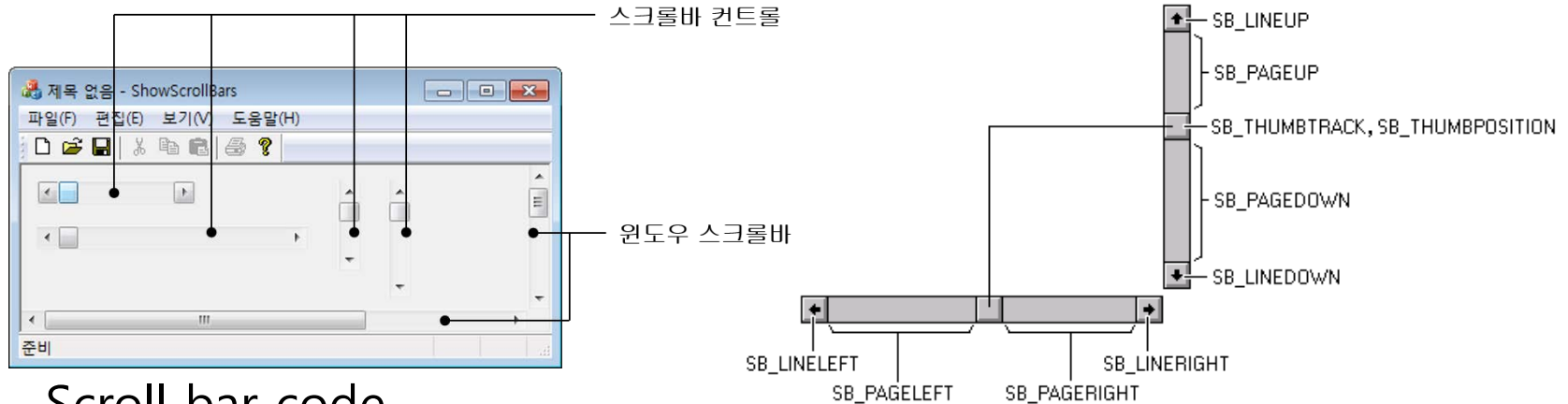
- 선택한 항목 얻기

```
int nIndex = m_combo.GetCursel();
if (nIndex != CB_ERR) {
    CString str;
    m_combo.GetLBText(nIndex, str);
}
```

Controls

- Scroll Bar

- 화면의 일정 영역을 scroll 하거나 정해진 범위의 값을 변경시킴



- Scroll bar code

- 사용자가 scroll하면 WM_HSCROLL 또는 WM_VSCROLL 발생
- 사용자의 구체적인 행위를 알 수 있는 정보(SB)가 같이 전달

Code

의미

SB_THUMBTRACK	Scroll box의 현재 위치로 이동 (마우스로 끌고 있을 때 계속 발생)
SB_THUMBPOSITION	Scroll box의 현재 위치로 이동 (마우스로 끌다가 놓을 때 한 번만 발생)
SB_LINELEFT, SB_LINERIGHT	왼쪽 또는 오른쪽으로 한 줄만큼 데이터를 scroll
SB_LINEUP, SB_LINEDOWN	위쪽 또는 아래쪽으로 한 줄만큼 데이터를 scroll

Controls

- Scroll bar message handler

```
void OnHScroll (UINT nSBCode, UINT nPos, CScrollBar* pScrollBar);  
void OnVScroll (UINT nSBCode, UINT nPos, CScrollBar* pScrollBar);
```

- nSBCode: Scroll bar code
- nPos; Scroll box의 위치
 - SB_THUMBPOSITION 또는 SB_THUMBTRACK인 경우에만 사용
- pScrollBar: Scroll bar를 조작에 필요한 CScrollBar class pointer
 - 윈도우 Scroll bar에서 발생한 것이면 NULL 값
 - CScrollBar 주요 member functions

Member	역할
SetScrollRange()	Scroll box 위치의 최소값과 최대값을 설정
SetScrollPos()	Scroll box의 현재 위치를 설정
GetScrollPos()	Scroll box의 현재 위치를 얻음

Controls

- Scroll bar message handler

```
void OnHScroll (UINT nSBCode, UINT nPos, CScrollBar* pScrollBar);  
void OnVScroll (UINT nSBCode, UINT nPos, CScrollBar* pScrollBar);
```

- nSBCode: Scroll bar code
- nPos; Scroll box의 위치
 - SB_THUMBPOSITION 또는 SB_THUMBTRACK인 경우에만 사용
- pScrollBar: Scroll bar를 조작에 필요한 CScrollBar class pointer
 - 윈도우 Scroll bar에서 발생한 것이면 NULL 값
 - CScrollBar 주요 member functions

Member	역할
SetScrollRange()	Scroll box 위치의 최소값과 최대값을 설정
SetScrollPos()	Scroll box의 현재 위치를 설정
GetScrollPos()	Scroll box의 현재 위치를 얻음

Controls

- Scroll bar 사용: 직접 Code 추가
 - CScrollBar 변수 추가

```
// CChildView.h
CScrollBar m_HBar;
CScrollBar m_VBar;
```

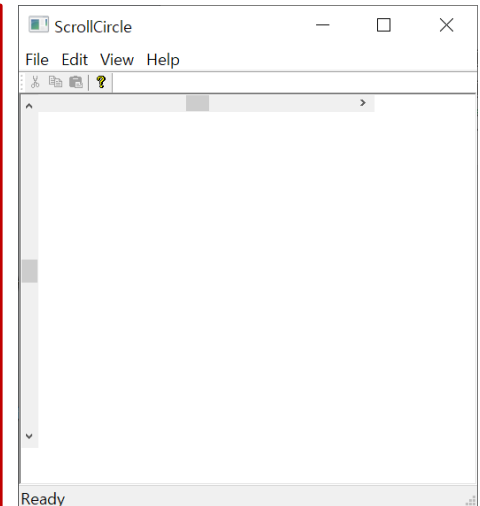
- Control bar 생성 및 box 범위와 위치 지정

```
// CChildView::OnCreate()
// 가로, 세로 scroll bar 생성
m_HBar.Create(SBS_HORZ, CRect(0, 0, 400, 20), this, 101);
m_VBar.Create(SBS_VERT, CRect(0, 0, 20, 400), this, 102);

m_HBar.ShowWindow(true);           // 가로 scroll bar 출력
m_VBar.ShowWindow(true);           // 세로 scroll bar 출력

m_HBar.SetScrollRange(0, 400);     // 가로 scroll bar 범위 지정
m_VBar.SetScrollRange(0, 400);     // 세로 scroll bar 범위 지정

m_HBar.SetScrollPos(200);           // 가로 scroll bar 현재 위치 지정
m_VBar.SetScrollPos(200);           // 세로 scroll bar 현재 위치 지정
```



Controls

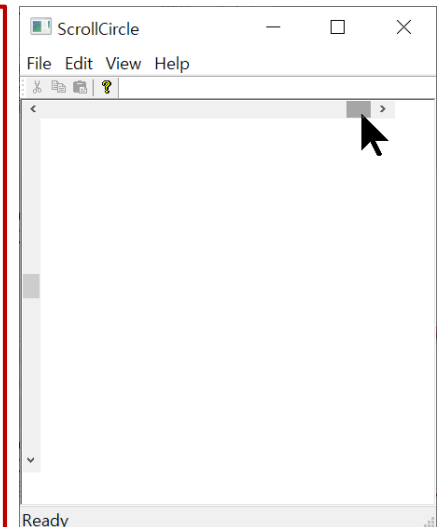
- Scroll bar 사용: 직접 Code 추가
 - MessageMap 추가
 - Scroll Bar는 ID에 상관없이 공통된 macro 함수 사용

```
// CChildView.cpp
BEGIN_MESSAGE_MAP(CChildView, CWnd)
...
ON_WM_VSCROLL()
ON_WM_HSCROLL()
END_MESSAGE_MAP()
```

– Event Message Handler 추가

```
// CChildView.cpp
void CChildView::OnVScroll(UINT nSBCode, UINT nPos, CScrollBar* pScrollBar)
{
    // Scroll bar의 box 위치 업데이트
    if (pScrollBar != NULL) {
        if (nSBCode == SB_THUMBTRACK || nSBCode == SB_THUMBPOSITION) {
            pScrollBar->SetScrollPos(nPos);
            this->Invalidate();
        }
    }
}

void CChildView::OnHScroll(UINT nSBCode, UINT nPos, CScrollBar* pScrollBar)
{
    // OnVScroll() 내용과 같음
}
```



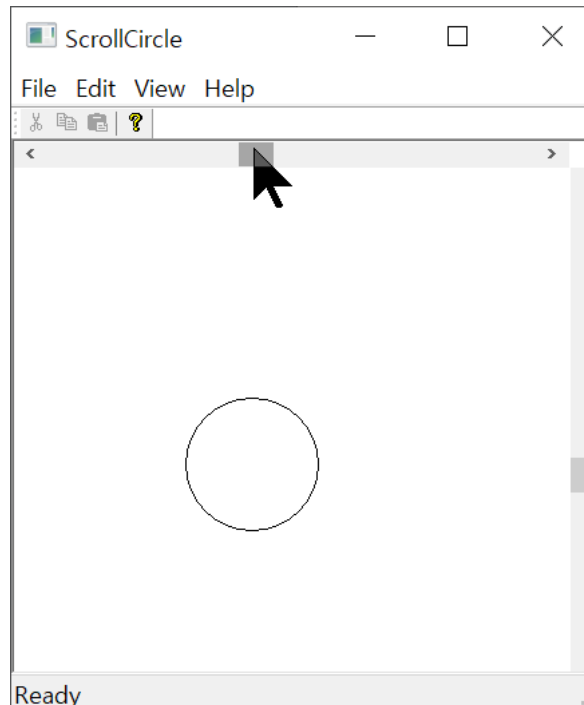
Controls

- 연습: Scroll bar를 사용하여 Edit 숫자 값 조절
 - Edit와 Scroll bar를 대화 상자에 추가
 - Scroll bar의 box를 움직이면 Edit에 그 값을 표시
 - Scroll bar의 range(범위): 0~99



Controls

- 연습: Scroll bar로 원의 위치 조절
 - 가로 및 세로 scroll bar를 각각 사용자 영역에 추가
 - Scroll bar의 크기를 사용자 영역 크기에 맞게 조절
 - `ChildView::OnSize()` 사용
 - Scroll Bar의 control을 움직이면 원의 (x, y) 좌표 이동



Q & A